Journal of Music Theory Pedagogy

Volume 2 Article 6

1-1-1988

Music Theory Software for the Macintosh

Bruce B. Campbell

Follow this and additional works at: https://digitalcollections.lipscomb.edu/jmtp

Recommended Citation
Campbell, Bruce B. (1988) "Music Theory Software for the Macintosh," *Journal of Music Theory Pedagogy*: Vol. 2, Article 6.
Available at: https://digitalcollections.lipscomb.edu/jmtp/vol2/iss1/6

This Review is brought to you for free and open access by Carolyn Wilson Digital Collections. It has been accepted for inclusion in Journal of Music Theory Pedagogy by an authorized editor of Carolyn Wilson Digital Collections.

MUSIC THEORY SOFTWARE FOR THE MACINTOSH

BRUCE B. CAMPBELL

REVIEW FORMAT

This review departs from the usual format in that several products, intended for quite different groups, are under consideration. Five are for ear training, and in keeping with the theme of this issue, they will receive the most comment. Two each deal with writing skills and "note processing." One product illustrates a new approach that has the potential for wide uses in higher education. The common element is the Apple Macintosh micro-computer. Therefore the review is more an overview of "repertoire for an instrument" (or attachments for an appliance, if you will) than a survey unified by topic. It is the author's hope that this review will help orient the present owner of a Macintosh in search of musical products, and also provide some insight to the person who owns another computer, or no computer, as to the special suitability of the Macintosh for musical applications. Finally, the information may aid a person or committee in charge of recommending to a college administration the purchase of microcomputers for a computer "laboratory" associated with a music department.

Despite its abrasion of literary sensibility, the review will make frequent use of the computer jargon that, for better or worse, has entered everyday speech.

ITEMS REVIEWED

A. Ear Training

1. *Listen* (version 2.0)

Resonate
P.O. Box 966
Menlo Park, CA 94026
(415) 323-5022
$99

Published by Carolyn Wilson Digital Collections, 1988

2. *Practica Musica* (version 1.51)

Ars Nova Software
P.O. Box 40629
Santa Barbara, CA 93140
(805) 564-2518
$129

3. *Ear-Training Expert* (version 3.0)

Techno-Arts
28 Daniel Street
Newton Center, MA 02159
(617) 964-0519
$150 single copy; $35 each in sets of ten

4. *MacGAMUT*

Mayfield Publishing Company
1240 Villa Street
Mountain View, CA 94041
(415) 960-3222
$15 (estimated)

5. *Melodic Dictator* (version 1.1)

Center for Performing Arts and Technology
The University of Michigan School of Music
1100 Baits Drive
Ann Arbor, MI 48109-2085
(313) 747-2020
$65
university "site license" available for $200

B. Part-Writing

1. *MacVoice* (version 2.0)

Kinko's Academic Courseware Exchange
4141 State Street
Santa Barbara, CA 93110
(800) 235-6919
$25.50

2. *Palestrina* (version 2.0)

pre-release version: will be available through
Kinko's Academic Courseware Exchange.
$20 (estimated)

C. Other

1. *Parsifal*

Jay Martin Anderson
807 West Lynn Street #201
Austin, TX 78703
(512) 476-3807
FREE (supply disk and SASE)

2. *Deluxe Music Construction Set* (version 2.0)

Electronic Arts
1820 Gateway Drive
San Mateo, CA 94404
(415) 572-2787
$ 99

3. *Professional Composer* (version 2.2)

Mark of the Unicorn
222 Third Street
Cambridge, MA 02142
(617) 576-2760
$495

ABOUT THE APPLE MACINTOSH

One of the important judgments a reviewer must make pertains to the idiomatic construction of the program, and its ease of use. To those of us who obtained the first incarnation of the Macintosh in early 1984, it seemed truly to be the computer "for the rest of us," especially for scholars in the arts or humanities who had earlier struggled with mandarin mainframes and mountains of punchcards in pursuit of relatively modest ends. The "bit-mapped" technology popularized by the Macintosh can be understood by

thinking of the computer screen as a very fine piece of graph paper. Each dot on the screen (or pixel—a contraction of "picture elements"), approximately 1/72 of an inch square, can be addressed by Cartesian coordinates. With its relatively fine resolution, the screen can represent graphical objects, including musical symbols, as easily as typescript. The dot-matrix Apple printer faithfully translated the pixels into ink dots on a page: the resulting WYSIWYG ("what you see is what you get") screen format soon became an industry standard.

The imaging is only one of many revolutionary attributes that the Macintosh brought to the consciousness of the casual public. Fundamental to its philosophy was the observation that human beings rarely operate in "modes," as computer interfaces have often demanded. In everyday life, when confronted with a decision that requires a "yes/no" answer, we may elect to put off the decision for awhile and do something else. (So-called "modeless dialogs" do exist in the Macintosh world, of course, but they present themselves in a distinctive double-framed rectangle that overlays the screen image as if on a separate plane.) The "mouse," a control device about the size of a bar of soap and equipped with a single button, permitted the operator access to any pixel simply by moving it around on a nearby flat surface. It gave the user a sense of ease, and more importantly, control over the machine—early advertisements stressed its "point-and-click" simplicity.

A frequent complaint from beginning users of more traditional computers was a lack of consistent behavior among programs. The letter "q" universally signals "quit" (exit the program), but many other letters have meanings unique to a particular program. To quite a few potential computer users, a "q" taken to mean anything besides "the letter q" results in confusion, and soon after, frustration. Actions that one wished the Macintosh to take were chosen from a visually isolated "menu bar" at the top of the screen in virtually every program. From the menu a list of actions, in simple English, could be made visible by clicking the mouse. One moved the mouse down the list to the desired action and released the button. This arrangement replaced the "command line" of earlier personal computers and prevented the kind of confusion mentioned above, for it structured the screen environment unmistakably.

The metaphor of a "desktop" encapsulated much of the design strategy of the machine. Referential terms with concrete, imaginative associations allowed the user to navigate relatively easily—for example, the "Clipboard" was a temporary storage location for data, the "Scrapbook" a more permanent disk-based one. In theory, at least, both could store any kind of data from any application and import it into another. Thus one could mix pictures among the text in a word processor, and instantly see

how the result would look on the page. A few years later, this principle led to "desktop publishing" (desktop is now used in its literal sense: one could devise publication-quality materials electronically and print them with the aid of a laser printer from atop one's own desk)—this journal is an example. The notion of "windows," or separate workspace modules devoted to components of a program lent additional visual orientation. One needed to make a window "active" by clicking the mouse button while the cursor was located on it before a particular set of responses could be invoked, a clever workaround the issue of "modes." The Macintosh even reversed the appearance of type on the screen, so that it is seen as black print on a white page.

To enforce consistency in appearance among commercial programs, the designers placed more than 500 computer instructions, or subroutines in "ROM" ("read only" memory) and promulgated "Human Interface Guidelines" in booklet form. Both encouraged program developers to adopt a similar "look and feel," so that the user could, as much as possible, employ the program "intuitively" (meaning, work with it successfully without bothering to read the accompanying manual). That two in-house products, MacWrite, a word processor, and MacPaint, a graphics application, were virtually the only reliable programs available for almost a year after the introduction of the machine contributed to the expectation of a "Maclike" way of doing things. (MacWrite is now often scorned by cognoscenti as "light duty" but it embodies the WYSIWYG, intuitive, and user-friendly criteria of the machine's fundamental philosophy better than most subsequent replacements to date. In setting up a computer lab in a college, one might keep in mind that a decision to provide the latest in complex software gadgetry may be self-defeating, because of the steep learning curve required to master it. By analogy, teenagers taking drivers' examinations for the first time will probably feel more comfortable with an automatic transmission than a stick shift.)

Very quickly, computer programmers discovered the "Law of the Conservation of Complexity": in essence, the easier the program is to use, the harder it is to write. This resulted in a delay in new products, and a fear among the "flannel-shirt crowd" that the Macintosh might be orphaned as had earlier pioneering machines. New memory chips and a rapid outpouring of new programs (called "applications" for the Macintosh) in 1985 helped narrow the gap between the original vision and its realization. The public and corporate America ceased to regard it as a toy and the rest, as they say, is history.

ABOUT EAR TRAINING

Before considering the programs themselves, and the place they may be given in an educational setting, we might consider what place ear-training itself has in the university curriculum and the education of a musician. One gets widely different answers, depending on whether one asks the performance instructor or the "music theory professional." To the performing musician, music theory is largely a means to an end—that end is simply, to hear better. This "outsider's" view of music theory might be summed up in a quotation from a recent review by Andrew Porter in the *New Yorker* magazine: "Elaborate analysis provides no certain help to musical understanding. Slogging through the *Journal of Music Theory* articles . . . will enhance . . . enjoyment . . . about as much as contemplating the circuit diagram of hi-fi equipment will enhance enjoyment of music proceeding from the loudspeakers."[1]

But many sensitive performers of contemporary music know and seem to care little about the technical compositional bases of the music they interpret. What *is* necessary is to play in tune, and to have some sense of how the music fits together as a whole. Formal analysis is shunned by scholars as "garden variety": detailed diagrams showing how phrases interrelate and work together, presented in a manner understandable to the interested performer, is rarely attempted by the music theory professoriate, and hardly a staple of "research" journals.

Porter goes on to say: "But technical knowledge in some degree is always helpful. At the least, [one] should know how to adjust the controls of his machine so as to yield the best result."

The knowledge needed to adjust the controls correctly, of course, is not gathered from technical manuals but from an attuned and alert ear. Of all those musical techniques for too long now condescendingly referred to as "skills" that are present in some degree in the undergraduate curriculum, ear training is the most important.

The interests of that student constituency known as "music majors" that the university curriculum must address at present goes well beyond the traditions and repertoire of 18th- and 19th-century Europe. Amidst competing partisanships and the resulting compromises, one might think the one common need of all musicians, be they aspiring educators, composers, popular performers, and even scholars—a sensitive and comprehending ear, would be the single objective best met by the university.

Rather, it is the most haphazardly approached of all musical subjects and the one of least status! The neglect of developing sensitive hearing has not been reversed by the establishment of Ph.D. programs in music theory.

These have erected artificial divisions between "pure" theory (hypothesis, actually), which is assigned formidable prestige, and the teaching of musical skills, deemed musicianship and regarded as unsystematic or unscholarly, and often relegated to persons lacking much formal training in theory. Many graduate students enter such programs with scanty systematic exposure to ear training, and as Teaching Assistants are only assigned to teach it (and presumably, learn it on the job) if they are deemed unfit to manage a section of lower-division theory. These students may never develop true expertise in musicianship, counterpoint, and voice-leading, prompting their faculty colleagues in performance later on to wonder what music theory really is, if it doesn't include musicianship.

The refusal to acknowledge the bifurcated nature of the discipline does considerable damage to the professional world of music theory. The field fails to give convincing evidence that the teaching of music theory requires special training, and so the myth is perpetuated that theory belongs on the lowest rung of the academic musical ladder, for anyone can teach it.

It is ironic that ear training is most successfully done in a traditional conservatory whose aim is clearly fixed on the European heritage, such as The Juilliard School. In the educational continuum between university on one side and conservatory on the other, the closer the curriculum gets to the former, the less effective is its ear-training program, quite often.

The ability to hear and to communicate what one has heard is hardly a vague skill. It can easily be tested, and such tests in other fields are the grist of standardization, and very often, "research" tools. This is where the computer comes in, of course. Where are the comprehensive data, compiled upon entrance and exit, of decades of university music majors? Where is the "reality-testing" applied to theories of perception and cognition, as is common in the social sciences?

Ear training is an extremely labor-intensive task. Like the learning of harmony or counterpoint itself, it is best done on the same one-to-one basis as a private lesson in an instrument or voice. It was a questionable curricular victory for "applied" music (the tapestry of meaning that the use of this formerly widespread term has is simply too rich and fascinating to explore further here) to debase music theory by consigning it to the classroom, often to a large lecture hall. No one can reverse the decision to offer theory instruction in colleges only in classrooms. But the thoughtful computer program can reclaim in some small way the calm patience and alert adjustment to the individual's need that mastery of the skill requires for many students.

ABOUT MIDI

Several programs permit the sending and receiving of data by means of the Musical Instrument Digital Interface. A synthesizer attached to the computer (along with additional hardware, a "MIDI interface") will play the examples and the user is free to choose whatever timbres the synthesizer affords. The programs permit a response to be sent to the computer from the keyboard as well. Those programs that have been on the market for a year or so have been retro-fitted with "MIDI capability" in response to the popularity of such hook-ups. A computer "lab" in a university should be directed and established by theorists, with pedagogical values foremost, and convenience (computers as musical typewriters) second.

It is too early to tell what impact computer-controlled synthesizers will have on "academic" composition. One thing is clear: the impetus for their recent development has come mainly from popular music. Before universities equate a "computer music studio" with a collection of commercial synthesizers and micro-computers (and make a large financial commitment), the question of just how it fits in with the education of musicians might be addressed. Universities have never done a very good job of teaching popular skills (such as, how to write greeting card verse) nor should they necessarily try.[2]

In many cases, the technological gleam and electronic puffery of the synthesizer leads to courses (even at the graduate level!) culturally comparable to beginning instruction in accordion or electric guitar. (Do we see English Departments offering courses in "word processing?") And where does the with-it music school find its funds for such a "computer music studio?" Why, through cutting back undergraduate ear-training and part-writing courses, naturally!

LISTEN

Sets of exercises from beginning to advanced levels in melodic dictation as well as interval and chord recognition are provided by this program. A unique feature is both a piano keyboard and guitar fretboard on which to input notes. The user can experiment with either on startup or by selecting "No Exercise." These are visible in Figure 1 (see end of article). The middle window that runs the length of the screen is the "Progress and Evaluation" window. It has three "pushbuttons" on the left (Start, Next, Repeat—keyboard equivalents are provided) that control the exercise, and a sentence or two of instruction. The elapsed time counter ("stopwatch") below them may be removed: it shows how long the current set of exercises has been

active. The squares over to the right from the stopwatch give information about progress of the current exercise. Each represents one note of the exercise. After the user has entered a note, the square corresponding to that note will change from the question mark to a check (correct) or "X" (incorrect). The two squares underneath the stopwatch provide reportage about the current set of exercises. The gray bars grow to the right as the number of correct or incorrect responses are tallied. (Outside of these bar graphs, incidentally, the program has no reporting capabilities.) There is no penalty for repeating the current exercise.

Communication to the user is also provided by the cursor, which takes on the shape of an ear (see Figure 1) when the user is to listen. It changes to a pointing hand when the user is to enter notes, and after the complete response, changes to a "happy" or "oops" face. The cursors are small, but they give a direct visual response and do not require the user to look away from the keyboard or fretboard.

The "Notation" window shows all the notes of the current exercise. This feature is of limited usefulness, because it displays the entire answer before one begins to respond. If kept hidden most of the time, however, it is useful as a reference if one gets stumped. The "Control" window allows the user to set four parameters: varying the "pace" and "duration" will result in legato or staccato notes. A group of twenty different timbres may be selected (most with fanciful titles): the current one is identified on the bottom line. The timbral choice is welcome in a long session, but as they are all produced by Macintosh software, they have a monotonous "electronic" similarity.

The sound output for the program is through the Macintosh speaker, or a pair or headphones. The program will also send and receive MIDI data so that an external synthesizer may be attached. Typical of the nice touches of this program, the three pushbuttons in the "Progress and Evaluation" window may be assigned to notes on the MIDI keyboard, so a set of exercises may be run conveniently without having to return to the mouse.

Separate sets of exercises exist for interval spelling and name, the latter by multiple-choice. Similarly, seventh chords may be identified by quality or may be spelled. Inversions of chords may be identified by multiple choice. Ninth, eleventh, and thirteenth chords (four notes, root position only) are also available, so the program has a special appeal for jazz and popular musicians. Further exercises include "random atonal chords" (unfortunately, never displayed in the "Notation" window) and "Growing Melody." This type of exercise begins with one note and grows as the user correctly identifies additions. If one makes an error, the melody is repeated, and only grows in length again after a correct response.

A menu item provides five levels of difficulty that apply to the

subsequent choice of any type of exercise. Also, a "User-Defined Level" causes the program to save to disk the parameter settings preferred by the user. This is a very useful feature. One can use the program at a later date, beginning exactly where one left off.

The user is given a very wide choice in defining the limits of the exercises. One can chose the range within which the pitches will fall, the starting note, the set of pitches available (for example, for "melodies," from one pitch class to the total chromatic) and whether intervals or chords will be "locked" (anchored by a user-specified pitch as top or bottom note) or appear anywhere in the chosen range. An "Auto-Select Materials" option will cause the program to cycle through a set of "dialog boxes" that permit adjustment of the content of each set of exercises. The option of "Sounding Answers" may be turned off, so instead of hearing the note of each key as it is clicked by the mouse (the only way to enter notes), one must wait until the entire melody has been entered before the program responds. This is similar to the classroom technique of dictation, where the student must take down a complete melody before receiving comment.

Listen is packaged with an excellent 86-page professionally written manual that explains the operation and features of the program in exemplary fashion.

From the music displayed in Figure 1, the reader can readily discern that this is primarily a pitch-recognition program. The "melodies" devised by the program appear to be at random from within the specified parameters. As was mentioned earlier, the durations of all notes are the same. These factors might appear to be inartistic limitations, but they strengthen the basic premise (identification and naming of simple musical phenomena) by excluding other considerations.

Listen fulfills its purpose admirably. It can be recommended to a motivated individual for valuable practice—for example, a graduate student who failed an incoming ear-training examination. Numerous copies in an ear-training "lab" should tempt diligent and persevering students to make significant progress.

PRACTIÇA MUSIÇA

This program is closer to emulating the experience of a college student than the preceding product because it contains a considerable amount of practice in what is often called the "basics" of music theory—spelling intervals and chords. Notes are entered primarily from the keyboard: the ingenious solution to enharmonic spelling is shown in Figure 2. For the sets of exercises in harmony, one may also click notes onto a staff.

SOFTWARE FOR THE MACINTOSH

An interesting feature is the possibility of hearing any exercise in one of nine different tuning systems. Another is digitized sound—the timbres of piano, organ, and harpsichord are astonishingly well reproduced. A consequent drawback is that the tempo cannot be altered except in the exercises in melodic dictation. Also, thanks to digitizing technology, rousing applause signals the successful completion of a level. MIDI data can be transmitted, and the user can input notes from a keyboard. The program has a "practice" mode that enables the user to experiment with the tuning systems and with notation. As notes are entered, they are classified immediately as to interval or chord, so the program functions analytically in a rudimentary fashion.

The activities available are divided into "Theory" and "Ear-training." In "Theory," one practices the spelling (and hearing) of intervals or chords. The user is free to indicate the notes in open or close position, and in any order. For example, faced with a request for a half-diminished chord on G in second inversion, the beginning student could write the chord in root position and then go on to interpret the "second-inversion" part of the question by removing the B-flat in close position and clicking a lower one. If the answer is partially correct, the program will "analyze" it as to root, type, and inversion, comment on which of these were correct, and show an acceptable solution in musical notation placed on the screen next to the student's response so they may be compared. Menu choices provide "Help" screens containing thumbnail synopses, and half of the manual (17 pages) is an "Introduction to Harmony." About these, however, the less said the better, insofar as current thought in music theory is concerned ("the triad... [is] two thirds stacked together")—they are on the same lamentably obsolete level as, say, Paul Harder's *Harmonic Materials in Tonal Music.*[3]

"Ear-training" provides six levels of difficulty in identifying intervals. One must identify the interval by name (multiple choice) and then click its second note on the keyboard. The beginning levels provide two choices of related intervals (fourths and fifths, for example) and proceed to add new ones as the user is successful. This tutorial approach is very effective.

Some of the algorithms that create the exercises could use refinement. In interval practice, this reviewer encountered the repetition of certain intervals with the same notes, even if they were answered correctly before. In the higher levels, very easy questions are found among the more advanced skills—frequently a waste of time.

The Melodic Dictation portion offers four levels of ten tunes each, ranging from "My Country, 'Tis of Thee" to Stravinsky's *Symphony of Psalms.* The program only checks for pitch accuracy—the correct note values are automatically made visible. A drawback is that there is no "authoring" capability for new melodies to be added to the pool. There are

Published by Carolyn Wilson Digital Collections, 1988

also four levels of computer-generated melodies. These are not completely random, but center around the tonic as specified by the user (except in chromatic exercises). The user may select key signature, scale (pentatonic through the total chromatic, including the church modes and the octotonic scale) and meter (including 5/4 and 7/8). Unfortunately, the melodies are played in a rhythmically expressionless manner. A useful feature is the "play first part" pushbutton (shown in Figure 2). Just a few of the notes are given when this option is selected. As one's notation of the melody progresses, the amount played is lengthened accordingly. The number of new notes beyond the notated portion may be easily changed. As the pushbuttons in Figure 2 indicate, the user can hear the entire melody at once, or hear a notated version of it.

Practica Musica has limited reporting capabilities. It will state the highest levels achieved in various categories and the number of melodies successfully entered. The several sets of exercises are called "games" and are scored in a possibly frustrating fashion: one must accumulate a certain number of points to complete a certain level. One cannot even try out a higher level until all lower ones have been successfully completed. In scoring, the program requires a certain number of consecutive correct answers, and it will subtract about half of the current score for a mistake. Pity the person who has gotten 29 points (out of a needed 30) for the "master level," only to be busted down to 15 after a false slip of the mouse! The decision to force the user along a single pedagogical path reminds one of fascistic "programmed texts": at least in a textbook, one can skip ahead! Adventurous users often try to find their own level through experimentation—only settling down to learn a program well after they have toyed with it, quite in the manner of a film composer acquaintance of the reviewer's, who, upon receiving the recording of Schoenberg's *Moses und Aron* and having never heard or seen it, turned straight to the "Orgy" scene.

Practica Musica can be very helpful in drilling students in the rudiments of chord spelling, and in gaining familiarity with the piano keyboard. The advantage that the computer has over any workbook is that the student can hear the notes when clicked, can hear the entire sonority, and if in error, can compare the answer both aurally and visually to the correct one. The ear-training melodies are well chosen and useful for students who have already mastered rudiments. Many of them will challenge a member of a second-year ear-training class. The computer-generated melodies are, if not really musical, at least conformant to many expectations of a coherent statement.

Lastly, it must be observed that seemingly no pains have been spared to make the product attractive and comfortable: a set of headphones is included; a practice pad of staff paper for dictation exercises; a short, but

complete, manual; and a handy "quick reference" card of intervals and chords. [Editor's note: I might add that the distributor for this product is unusually cooperative in responding to communication, providing update information, etc.]

EAR-TRAINING EXPERT

This package consists of two disks, a manual describing the programs and a textbook/workbook. Outwardly, it seems to have been designed with the college undergraduate in mind. This explains the pricing structure, which encourages purchases in quantity by a bookstore or other student outlet. The authors use the term "softext" as a combination of "software" and "textbook."

The organization of the textbook portion is problematic. Chapter 1, "The Nature of Music," discusses the physical properties of sound with reference to sine wave, amplitude, frequency, and period. This material is irrelevant to beginning ear training, and could be left out or placed in an appendix. The chapter introduces three small programs on Disk 2 that demonstrate manipulations of waveforms. These seem to have been written by the authors to learn Macintosh programming. The programs do not conform to elementary expectations of the Macintosh Interface and they do not even stand alone—they require a BASIC run-time module. They reminded this reviewer of the "freeware" widely circulated among users' groups back in 1984-85; they appear embarrassingly unsophisticated today and should be rewritten into a single program or discarded. An irritating quirk is the pompous term "End Session" instead of the universal "Quit."

That the technicalities of the first chapter seem misplaced is evident by turning to the second, which introduces terms such as "the staff," and what a clef is. All the C clefs are described before the treble and bass are mentioned. Subsequent chapters are devoted to meter, scales, key signatures, intervals, triads, and chords. These read as if the authors simply sat down at the word processor and hastily typed in their recollections, grouping things in the laziest way, by topic. A few useful tips are included, such as those mnemonic gimmicks a Teaching Assistant might give out. In a college situation, all such material would be covered in the first semester of elementary theory.

These examples point up the difficulty of trying to determine for whom the book is intended. The textbook portion is so rudimentary that it seems aimed at the "teach-yourself" market, but the workbook portion implies an organized classroom approach. The workbook is roughly equal in length to the textbook portion, and contains questions reminiscent of

those "programmed texts" that exist mostly for the sake of multiple choice tests, such as "what is melody?" to which the following should be regurgitated: "melody is a succession of pitches which is perceived by the mind as a unity." This is hardly something that anyone can consider meaningful music theory any more. The workbook also contains notation and identification exercises for the topics in the textbook portion. These would have to be corrected by an instructor. One of the advantages of computer-based learning is that students can "hop around": they can acquire confidence in the consonances and then proceed to triads and their inversions, rather than trying to master all the intervals before continuing on to triads and chords. In fact, many student can differentiate major and minor chords and yet have trouble identifying their intervals.

The authors could have dispensed with the "softext" idea altogether if they had used much of the workbook as a blueprint for additional features in their computer programs. For instance, there is a worksheet devoted to clef practice: "Transpose the following notes to the various clefs." (One questions whether "transposition" is the correct term for rewriting a passage in another clef—such sloppy usage is not isolated.) The computer is very good at this type of simple drill, and students benefit from immediate response. Besides, it is terribly tedious to correct such exercises—ask any Teaching Assistant!

The main pedagogical material is found in two separate programs, "Tutor" and "Drills." Neither offer musical notation, a serious flaw, nor any grouping of exercises by level of difficulty. The former offers ear-training exercises in major and minor scales; church modes; the pentatonic, whole-tone, and chromatic scales. The user can choose the length of the excerpt (1 to 100 notes) and the tempo. Also, one can select a key, or let the computer choose. (It must be noted that many of these segments stretch extravagantly the authors' definition of melody cited above.) The user must click keys on the screen representation of a keyboard to enter notes. All notes are equal in duration. The program will only determine if the user clicked notes in the same sequence as they were played by the program. The process of entering longer segments can be quite confusing. Curiously, the textbook offers little advice in how to use the program. It would be this reviewer's recommendation that the student take down the dictation on music paper completely before proceeding to the separate task of entering the response, which may be trickier. A pushbutton, "Display My Response," will play back the student's answer as the corresponding keys flash. This implementation appears to be inconsistent with the textbook, which emphasizes notation.

"Drills" is an identification program for scales, intervals, triads, and seventh chords (arpeggiated or simultaneous). The user can select how many different kinds of each type should be included. There is no way to

mix in the same set of exercises, say, intervals that ascend or descend from the first pitch given. Lacking perfect pitch, the user has no way of knowing what notes are being played—the program tests for interval or chord "quality" only. (While there are probably still some instructors left who teach in a likewise fashion, the program should be enriched with the possibility of knowing what notes are played, to help develop the student's relative pitch.) The user responds by multiple-choice pushbuttons. Unfortunately, those types the user has left unselected are not dimmed or removed. The program suffers from a cumbersome interface: one can't even use the menu bar when a specific set of exercises is active.

Both programs provide very detailed reports. These could serve to prove to an instructor that a student had really done the homework, by practicing an assigned drill. (The crafty student, however, could save a report on disk, and modify it.) If there were a statement by the authors of how the report might be used, the data, which tracks nearly every click of the mouse, might be more pertinent.

Ear-Training Expert seems to be a rough draft. The programs are so slight that the product probably couldn't be taken seriously without the accompanying book, which contains an unimaginative rehash of what might have passed for basic theory 30 years ago—the bad old days when music "theory" was music "fact." Programming gaucheries aside, the authors lack a fresh perspective on the materials they present. Especially telling is the lack of insightful organization about how human beings actually learn and process what they hear. This could signify an insufficient background in contemporary music theory, the corrosive effect of programming in BASIC, or both.

MacGAMUT

This product was furnished to the reviewer in an unfinished state ("beta-test," in the jargon). Therefore some of the deficiencies mentioned below may have been corrected by the time commercial distribution is in place.

This program (an acronym for Graded Aural MUsic Training on the Macintosh) drills beginning students on basic musical materials (intervals, scales, chords). It was developed at The Ohio State University, where it has been tested on more than 300 students. Some of its idiosyncrasies stem from the constraints under which it was originally written. An administrative decision resulted in Macintoshes being set up in a lab without keyboards, so the program is entirely "mouse-driven."

The product actually consists of six parts: "MacGAMUT start," which

is a kind of traffic controller; separate programs entitled, "Intervals," "Scales," and "Chords"; and two others. If one double-clicks directly on one of the last three named programs, a message will appear telling the user to click on "MacGAMUT start" to begin work. This is eyebrow-raising, to say the least, for anyone used to the "look and feel" of commercial programs. A much tidier solution would be a single program icon that would later branch into the three areas of drill. "Get Stats" and "Set Params" are programs whose separate existence is justified. The former will provide statistical information on a student file: current level, total numbers correct, the number tried, and the total time spent on each of the three drills. The information could be made available in a visually less demanding format, but the idea is a good one. (It is easy to imagine the following scenario: each student submits a disk at the beginning of class and a Teaching Assistant processes each one with a revised form of this program designed to accumulate data for the entire class, and display it in charts and bar graphs. This kind of rapid analysis would provide valuable information in setting up the next ear-training class session.) "Set Params" need not appear on a student's disk at all. It will adjust the options available to the student in the drill programs. The instructor can set the number of correct answers needed for "mastery" of a particular level, how many chances the student will have to answer correctly, whether the student can choose the tempo, and what clefs will be used for display. One hopes that a revised form of "Set Params" will be forthcoming that will automate this task, so that a Teaching Assistant could feed rapidly an entire class's worth of disks into the computer, setting each disk's parameters identically. Figure 3 is a screen shot from "Set Params," and Figure 4 is a sample of the dialog window presented to the student. (Unavailable choices are dimmed.) The care with which the learning structure has been devised is evident. It should be noted that the instructor is free to rearrange the order of levels present in Figure 4 to meet particular needs.

Each module (intervals, scales, chords) combines listening and notation. For example, in "Scales," at first the student is asked to enter the entire scale (including accidentals) by clicking and "dragging" (moving the mouse while simultaneously depressing its button.) Higher levels present the scale already on the screen, and ask for the accidentals to be provided. The module tests for the major scale and the three forms of the minor scale. The "choices" mentioned in Figure 4 refer to the number of multiple choices offered. For instance, level one (intervals) asks the student to choose among "m3, m6, M7." This will build confidence with beginners. In the more advanced levels, the program will accept different answers when the range of choices make it possible; for instance, an interval of 4 semitones can be labeled as a "d4" or a "M3." If the student has responded incorrectly after

the specified number of attempts, the correctly notated solution appears on the screen alongside the student's effort, along with three additional push-buttons: "Play Notes" (those that the student entered), "Play I.D." (the name that the student chose) and "Play Correct." Incidentally, it is pleasing to note that for questions about triads, the user is asked to click "5/3, 6/3 or 6/4" rather than "root, first inversion, second inversion" thus underlining the intervallic constituency of the triad.

Reservations this reviewer has about *MacGAMUT* deal mostly with its appearance and operation. After all, a person used to driving a car expects the accelerator pedal and gearshift to be in fairly standardized locations. For a motorcyclist, the positions are different, so it is less a matter of logic than habit and the safety and convenience that accompanies it. The standard "Apple," "File," and "Edit" menus are not supported (see the menubar of Figures 3 and 4). The music notation is large and exceptionally clear, but the pushbuttons are occasionally uncomfortably close together. Sometimes these buttons are inoperative without a visual cue and occasionally they disappear (dimmed buttons is the expectation in both cases). An on-screen tutorial appears in reverse video if one has selected "Get Instructions" from Figure 4. It has the slight advantage of permitting the user to try out some of the buttons while in this mode, but it is bewildering and "anti-Mac" in its layout. Instructions should be available as a menu item, and the text should appear in a separate scrollable window. The program fulfills its purpose admirably, but lacks panache. It is somewhat fragile, too: this reviewer encountered numerous system errors.

The pedagogical basis of the program, however, is excellent. It could be integrated very smoothly into a standard college course in beginning ear training. It has the advantage of teaching notation at the same time. Certainly it is foolish to separate written and aural skills at beginning stages, and *MacGAMUT* encourages a unified classroom approach. An instructor might want even more flexibility in arranging the levels, however. Usually the higher levels incorporate materials already mastered, For example, one may wish that "major-minor" and "major-major" chords could be left out of a level so there would be more drill in differentiating "minor-minor," "diminished-minor," and "diminished-diminished" chords. Similarly, it is a waste of time to be presented with a unison or a perfect octave at a high-numbered level. There is no MIDI capability as yet, but it is promised soon.

MELODIC DICTATOR

In August 1987, Apple Computer introduced a new software package called *HyperCard* written by Bill Atkinson, the programming whiz who

Published by Carolyn Wilson Digital Collections, 1988

developed *MacPaint*, the revolutionary graphics program that came bundled with the original Macintosh in 1984. The company has thought highly enough of *HyperCard* to have it included with all Macintosh computers sold since then, encouraging its acceptance as system software. Attempts to define this new product have stymied the popular press: Atkinson himself is reputed to have referred to it as a "software erector set." Perhaps "unstructured information management program" is as close as any other definition. Central to the concept of the software is the possible customization by the user through levels of program control ("authoring" and "scripting") in its own interpreted language, HyperTalk. In this way, the terrifying steep learning curve for mastery of Macintosh programming is circumvented to a considerable degree: HyperTalk becomes the software equivalent of the Macintosh itself as the programming language "for the rest of us." Speaking generally, herein lies its main drawback: persons with no background in programming can assemble a visually attractive format in much the same manner that a person with no musicality can draw many varied sounds from a "state-of-the-art" (the irony!) synthesizer. *HyperCard* lowers the barriers to programming, but, as Ansel Adams was reported to have said, the ability to operate a camera does not necessarily make one a photographer.

Melodic Dictator appears to be the first example of a pedagogical program in music developed with *HyperCard*. The version furnished for review was an advanced "beta-test" copy. It is accompanied by MIDI *Dictator*, a variant that will play the exercise on a MIDI device and will accept input from a MIDI keyboard. They were written by David Gregory, director of the Center for Performing Arts and Technology at the University of Michigan. The program has a template file (files are called "stacks" in HyperTalk, a metaphor deriving from a stack of cards, as used in a Rolodex file) for use by an "author" to create "student stacks." These stacks can then be distributed without the component of the program that authored them. Therefore a consideration of the program must have two viewpoints: one of the instructor responsible for putting the student stack together, and another of the student, or "end-user."

Melodic Dictator is designed to be used in a university setting, for some individual (a professor or Teaching Assistant, more likely) must devise a set of 24 single-line dictation exercises (no more, no less) and enter them into the computer. Each line may have up to a maximum of 15 notes. Various note values may be used. The premise is admirable—the instructor has control over what students hear. As instructors entering notes progress from melody to melody, they can build on the earlier ones by adding, changing, or inserting notes (provided that the clef is not changed). This feature makes *Melodic Dictator* an actual teaching tool, not just a repetitive testing machine.

One could establish certain structural intervals (e.g., the perfect fifth) and lead the student to hear others, such as a minor sixth, as the structural interval plus a semitone, rather than as something in-and-of-itself, which it rarely is in tonal music.

However, tedious is too kind a word for the editorially inflexible and dreadfully slow process of entering the notes for the exercises (this reviewer used a Macintosh SE with an internal hard disk). The combination of the "pointing finger" cursor (see Figure 5) and the narrow spacing of the staff lines (five pixels) results in frequent input errors. This reviewer found on several occasions that a situation arose that required quitting the program in the midst of assembling a student stack. Without a detailed knowledge of *HyperCard* programming, such misfortunes mean starting all over again. Since it takes at least an hour to enter 24 melodies of respectable phrase length, the tedium quickly reaches maddening proportions. Here is clearly a case where an attempt visually to mimic a real application in *HyperCard* defeats the intent of the Macintosh (ease of use). As the authoring component is not meant to be seen by the student, it could be altered to accept and translate a simple text file. Then, using a simple input code (for the range is limited to two octaves, and the note values to 10), the instructor could employ any word processor to edit an input file, which would remain in a convenient form for future modification. A small subset of DARMS would be appropriate.[4]

Students are presented with a screen similar to that in Figure 5. After listening to the example played by a digitized instrument (no limit to the number of times), students must click on a note value, and then click on the line or space of the staff to enter the answer. They cannot change the tempo, and the program cannot play the students' response so they may compare it with the correct solution. Entering notes is just as slow and tedious for them as for the instructor-author. There are no reporting capabilities. Obviously, the notation is confusing and unclear—notes are not centered squarely on lines or spaces, and stems always point upwards. These are not habits we wish to encourage among undergraduates. Numerous "bugs" still afflict the program: one had the effect of turning the cursor into a "drawing tool"; another turned on "text insertion" when the user hoped to enter additional notes; and a click on the "Help" button while the answer was present (obtained by click "Check") found the answer floating disconcertingly on top of the "Help window."

MIDI *Dictator* is a more successful and flexible program. The notation is much faster and clearer. One can range over the entire grand staff. The instructor can determine in advance how many exercises will be in a student stack, and how many opportunities the student may have to hear the melody. After it is sounded from the synthesizer, the program goes

immediately into "record" mode so students may play it back without delay. Unfortunately, the program does not test for rhythm: it permits the quarter note value only. When students select "check," the response appears on the screen along with a comment telling how many of the notes were played correctly. At the end of a set of exercises, a cumulative score will be given. The program will not recognize as partially correct an instance where the student played back the exercise intact, but added or omitted a note at the beginning. A more sophisticated checking algorithm would place the intervals of the exercise into an array, and make several comparisons with the student's response. As the classroom instructor knows, frequently a student will get one interval wrong in a dictation, but most of them correct. At a glance, the instructor can see that one misapprehended interval might cause the remainder of a student's melody to lack the correct pitches. However, most of the exercise might be correct if viewed as a string of intervals. More attention to the pedagogy of ear training as a "research" field might result in programs that assist the student in such ways.

Both programs are valuable in that they give the instructor the capacity to design all the exercises. They point in the direction that useful ear-training programs might go, for they allow instructors to make many of the same choices they would make in choosing exercises for a Teaching Assistant to use for drill in a lab session. *Melodic Dictator* may be summed up as a tempting prototype. *HyperCard* is too clumsy a choice for its realization: a re-working as a compiled program with flexible editing and input facilities is needed before its widespread acceptance can be foreseen. One can envision ready-made student stacks that would be accompanied by a printed copy of the melodies present in each, for selection by the instructor. While this would reduce the unacceptable level of boredom that putting together a stack presently takes, it would vitiate the most valuable feature of the program's design, the opportunity for customized authorship.

PARSIFAL

This program was not designed as a pedagogical tool, but it is included for review because it shows the ways in which *HyperCard* may be used as a database that can branch effortlessly in many directions. It is in "continuous development" by its author, an amateur opera lover. While in its current form it displays opportunities for further refinement, it is nevertheless a good and imaginative introduction to the way in which *HyperCard* might be used as an undergraduate instructional aid. A "home screen" for the stack presents eight topics (Characters, Interpretations, Leitmotifs,

Production, Sources, the Story, Recordings, and References). The stack builds on the inherent strengths of *HyperCard* to link data through multiple cross-referencing. For instance, "Characters" presents the user with a list: by clicking on keywords (prefixed by "•") within "cards" assigned to the various personages one can hear an associated leitmotiv by means of digitized sound, see a digitized picture of an opera singer costumed as the character, and so on. "Production" links commentary to pictures of stage sets.

The nature of *HyperCard* is evolutionary: any user can easily add bibliographic data or update various parts of the program with additional text, pictures, and sound. While the concept can be compared with a set of class handouts, the stack can be much more easily modified and enriched, and simple animation could demonstrate, for example, stage blocking in a way impossible on paper. Seeing or hearing similarity in seemingly dissimilar passages—the creation of links and references—is an important goal of music analysis, and a well thought out *HyperCard* can help tremendously. One can foresee the widespread distribution of such electronic "handouts" in the near future.

MacVoice

This program offers assistance in part-writing: four-part chorale-style exercises, the basis of much elementary study. Figure 6 shows the basic layout of the program. Up to 50 chords may be presented in a scrolling (left-to-right) window. Entry of notes is very fast, and editing is straightforward by means of special cursor shapes. The staff lines are seven pixels apart. Voice crossing is possible. A roman-numeral analysis is instantaneously made below the bass staff. All chords are interpreted in terms of the scale degrees of the tonic key, which is determined by the key signature as selected from the "Settings" menu, so a modulating passage will not be analyzed as such. An example can be heard through the Macintosh speaker, and it may be saved as a file.

MacVoice is very well written, from a programming standpoint, but weaknesses abound in the domain of music theory. Some questionable errors can be seen in Figure 6. The program will cite the progression of scales degrees 8-7-6 harmonized with I-iii-IV as incorrect ("leading tone doesn't go up to tonic . . ."). The pattern 6-7-8 harmonized as IV-vii°-I is incorrect if the augmented fourth in the middle chord ascends to a perfect fourth. In a root-position dominant-seventh chord, it is wrong to leave out the fifth, even if the root is doubled. In some instances. the program did not spot a root-position dominant-seventh chord whose seventh was led upward to

the fifth of a tonic chord in 6-3 position. So-called "stationary octaves" (between two perfect intervals, one pitch stays the same and the other leaps an octave) are found to be "parallel octaves or unisons." This reviewer found that every case of a triad with a doubled third was considered incorrect. No rhythmic advice is offered, even as to the alternation of strong and weak beats (which could be rather simply accomplished, as chords are numbered odd/even), and so abused cadential 6-4 chords will not be caught.

The concept is excellent: the program is analogous to a "spell checker," which is a feature of many word processing programs. But *MacVoice* is disgracefully simplistic. It exhibits far too many of the faults associated with unimaginative teaching of part-writing: too much reliance upon too few and over-generalized rules, and ignorance of the practice of musical literature. It is a rule-based rather than a conceptual approach, and it seems to ignore the strides that have been made in basic harmony textbooks of the past 15 years, let alone the "research" side of the profession. The program could help a student achieve a very narrowly cast kind of correctness, one unconnected with the practice of most music. Its appeal seems limited to the dull, methodical student who might later become the kind of theory teacher who finds satisfaction—even pleasure—in placing a roman numeral below every verticality (be it Bach, Beethoven, Debussy, or George Crumb).

Programs of this type hold great promise, however, as in many cases the part-writing component of basic harmony classes has been diminished due to the labor-intensive nature of correcting student papers. The areas of artificial intelligence and "expert systems" offer significant opportunities for research topics.

PALESTRINA

This program was developed by David Evan Jones at Dartmouth College, a member of the "Apple University Consortium." Dartmouth is the source of many interesting and high-quality programs for undergraduate use in a variety of fields. *Palestrina* checks voice leading in the area of sixteenth-century sacred vocal polyphony (the "Palestrina style"), as understood by Knud Jeppesen.[5] The program will examine two-part exercises in first, second, and fourth species. Up to 12 notes are permitted in the Cantus Firmus. The user may choose from a variety of clefs. The screen allows one to enter a line both above and below a given cantus, and although they are understood to be independent of each other, the program will play them simultaneously if the checkboxes above and to the left of each clef are activated. Errors in the line are marked with an "X" above the staff. When

one clicks the "X," it is then surrounded by a square, and a comment appears in the message rectangle near the bottom of the screen. A comprehensive series of "help screens" is available at all times from the menu item "Rules" (see Figure 7): an error comment is preceded by a number so the student may then proceed to the appropriate screen for further guidance.

Use of this program permits the student to turn in virtually error-free counterpoints so that class time can concentrate on aesthetic refinements. The printed copy will be welcome to the instructor who has become weary of sloppy assignments. The program comes with seven files of cantus firmi and 12 fully worked out exercises, so the student may become familiar aurally with the expectations of the style. The ability to add barlines (*Mensurstrich)* is a nice feature, although a global on/off for barlines would be more elegant and useful.

Jeppesen's observations about "the striking difference in the treatment of ascending and descending movement"[6] is an important point of style that the program does not address well. Nevertheless, *Palestrina* is a very fine program and should make counterpoint study more productive and efficient. One looks forward to future programs from the author for third and fifth species, and for counterpoint in three and four parts.

Ironically, students and many instructors unburdened with a sense of the history of music theory have come to believe that "traditional" music theory study is basically the study of harmony from a book such as Piston's.[7] With programs like *Palestrina*, species counterpoint may regain, in the pedagogy of tonal music, the former noble position it held in the eighteenth and nineteenth centuries. Eclipsing the conceptually bankrupt "direct approach" would be an added plus.

MUSICAL GRAPHICS

Space does not permit detailed comment about "note processors" but the interested reader is directed to two for further investigation. Provided that one does not need a system with more than eight staves, *Deluxe Music Construction Set* is a good choice. It permits a more flexible approach to editing than its expensive cousin, *Professional Composer*. As it is graphically based, it is better generally for piano music than *Professional Composer*. Its staves are 20% further apart, too (five pixels; *Composer*: four), so for brief examples and handouts, the instructor need go no further. It will also handle large compositions. It provides direct MIDI output, and it supports the "Sonata" musical font developed by Adobe Systems for high resolution printing on the Apple LaserWriter and related printers.[8]

Published by Carolyn Wilson Digital Collections, 1988

Professional Composer will handle very complex scores, and is best suited to music for instruments that perform one note at a time. Complicated piano music requires each line with a different rhythm to be entered on a separate staff, then merged. In theory, at least, the program will make a piano reduction from a large score, and also extract and transpose parts. Meticulous musicians will find that the program does not quite live up to its claims. For instance, the program controls how many measures appear per system: overriding its preferences must be done as a system-by-system workaround. The latest version of the program supports the "Sonata" font. A related product from the same company, *Professional Performer*, is a sequencing MIDI program. A feature that will be of interest to readers of this journal is its ability to export files to *Composer*. Therefore, one may play a piece at the MIDI keyboard under *Performer*, then switch to *Composer* and have it appear in musical notation, where it can be refined—clefs changed, lines transposed, or note values altered. For someone with keyboard proficiency, this is the fastest way to enter a score. A host of pedagogical tasks can be made easier thereby, from the preparation of worksheets for freshman theory to the making of performance editions in a graduate seminar.

GENERAL OBSERVATIONS

Several of the programs use a "random" method of generating pitches or for choosing intervals, or chords. Some sort of "memory" should be built into the programs so that one doesn't get the same type of example too often, especially easy ones. For example, in the hardest level of a "chord-quality" identification program, this reviewer was presented with three major root position chords in a row! Many of the programs employ some form of copy protection, which can be a headache for the user of a hard disk. More sophisticated algorithms need to be written, especially to handle rhythm, and systematic "research" needs to be done in learning theory, perception, and cognition.

It is the music theory component of many of these programs that is weakest or obsolete. That is not completely surprising, as only a few first-rate institutions seem to recognize the connection between the latest "research" and the classroom (as is taken for granted in many other fields) and demand trained theorists. "Research" as would impinge upon ear training is not forthcoming due to the low esteem in which it is held, and even at such places ear training is often taught at a level far below that of musical analysis. It is not surprising to learn that several programs reviewed here are the work of former graduate students in music who applied their

programming knowledge to the chore after a year or so in the classroom. Sometimes one felt that an author had an urge to program something musical, cast about for topics, and not too surprisingly landed on elementary ear training. When as much (or more!) effort goes into the central task of *what* to program as has gone into mastering the complexities of the Macintosh interface, then we can expect products that will take computer-assisted instruction to a new plateau.

Above all, trained music theorists should become engaged in program design, and in the study of perception. The field of music theory should award a new dignity to such studies: presently it still confers pariah status on topics remotely pedagogical. There should be much more dialog about the goals of ear training, and more knowledge needs to be gained about the "mind training" that is so crucial a part of sensitive hearing.

SUMMARY

Computer-assisted instruction is still in its infancy. Much of it is automated routine testing. But when one compares most of the programs reviewed above to such a simple program as *Melodious Dictator* for the Apple II[9] (still available, and for $150), one sees just how far things have come in a very short time. These programs afford the motivated student an opportunity to attain a level of proficiency that most courses cannot match. They point to a future of increasingly powerful products that will truly enrich the study of basic musicianship, and thereby lay the foundations for increased musical sensitivity.

Figure 1. Listen.

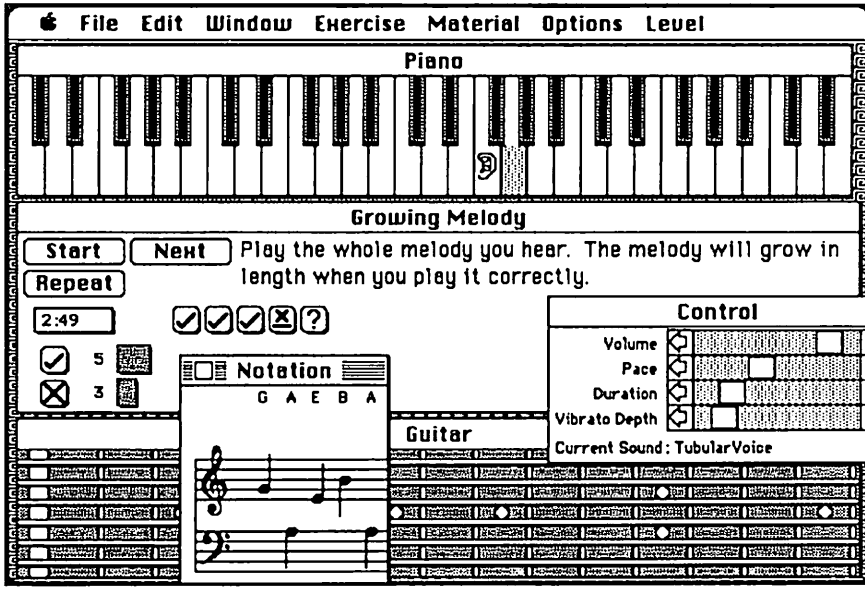


Figure 2. Practica Musica.

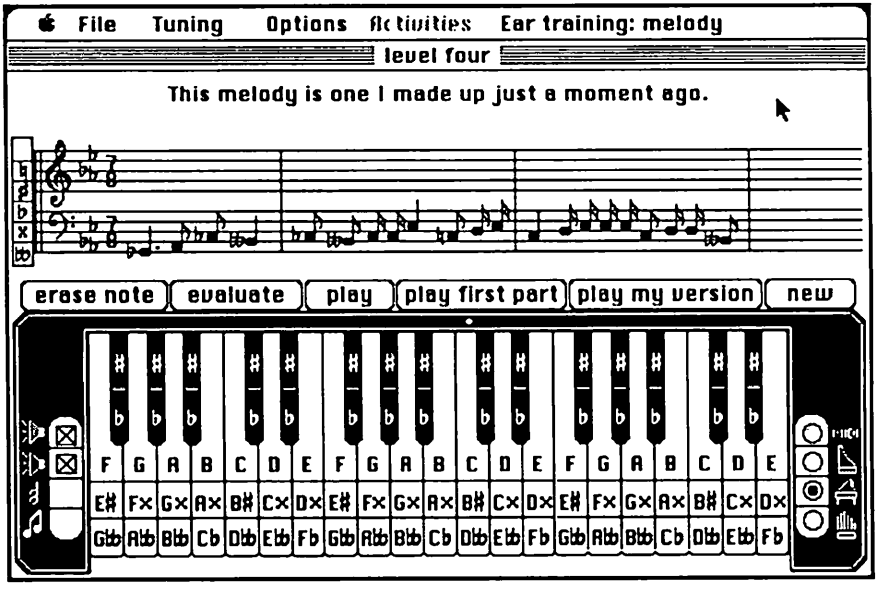


Figure 3. MacGAMUT.

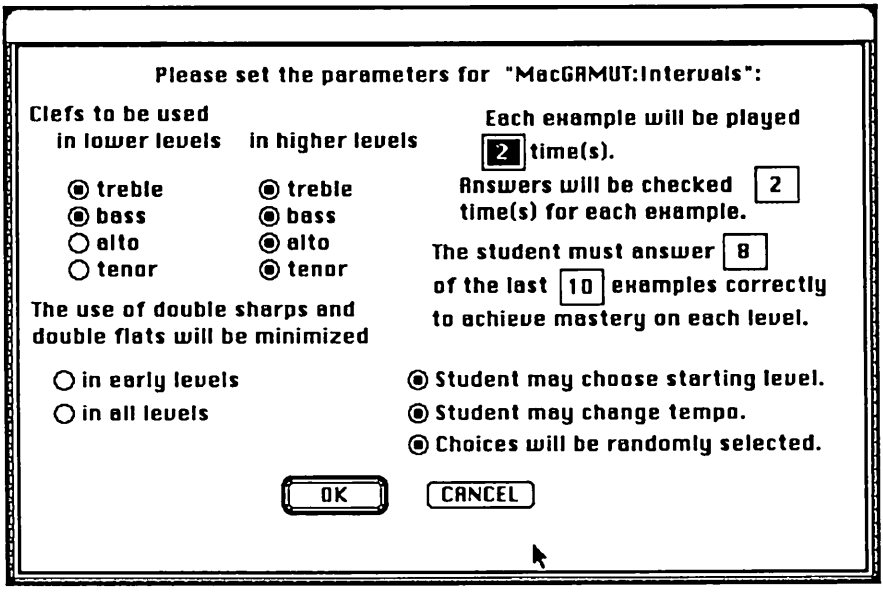


Figure 4. MacGAMUT.

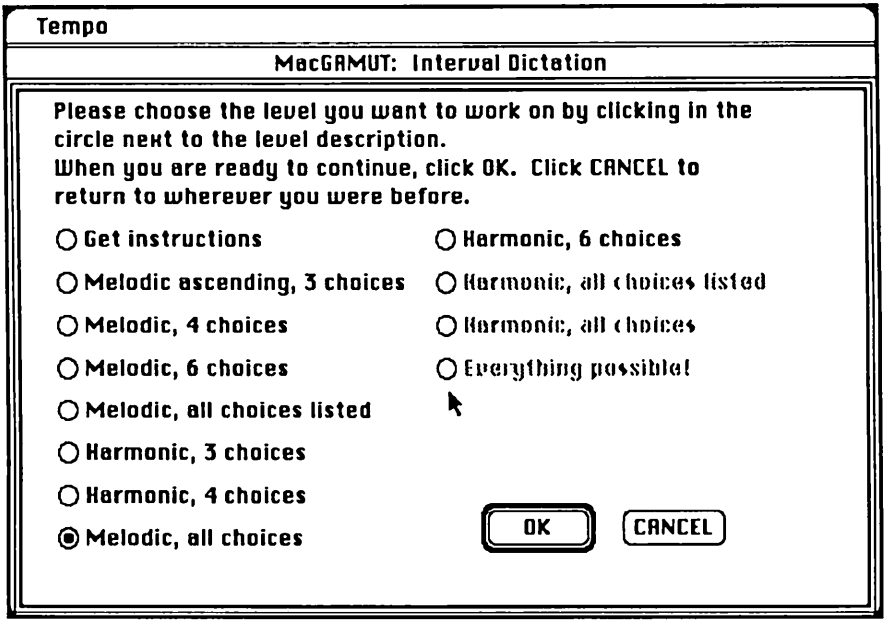


Figure 5. Melodic Dictator.

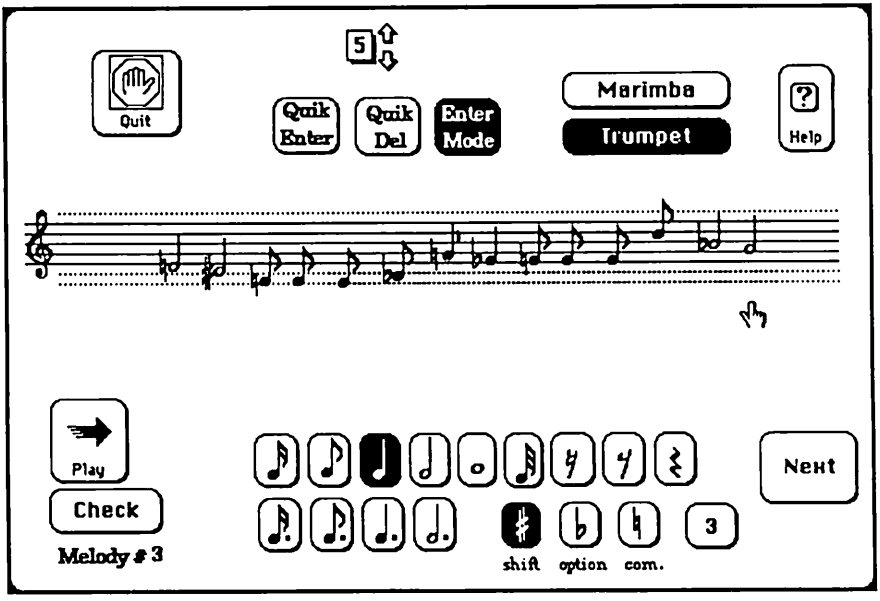


Figure 6. MacVoice.

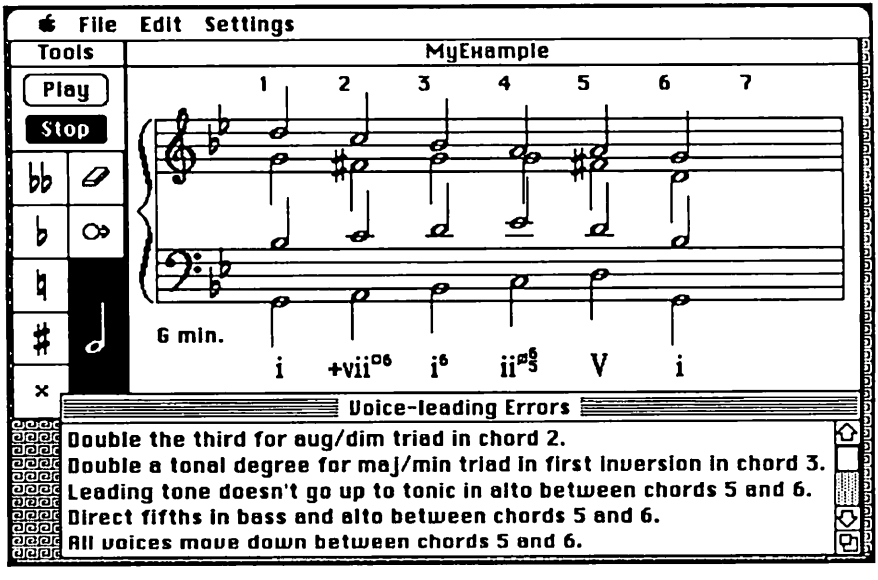
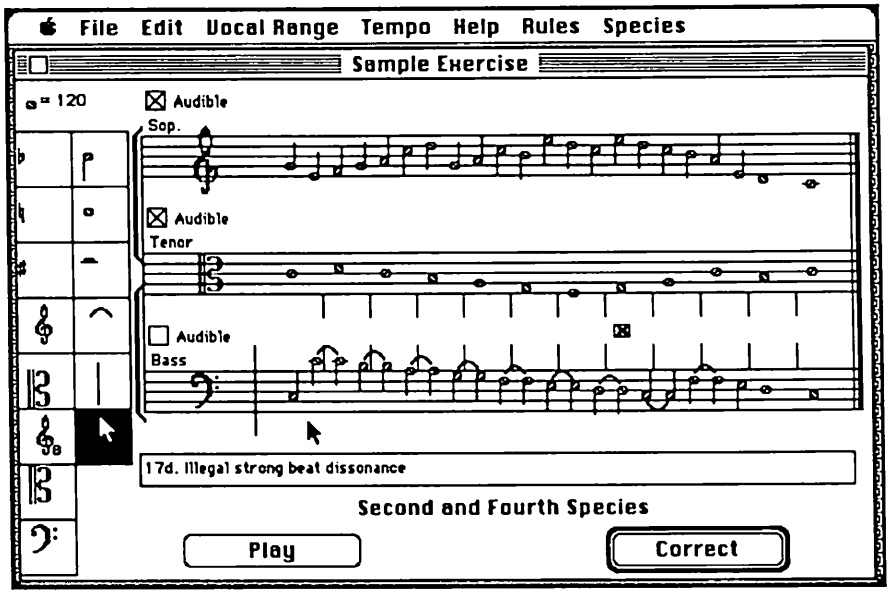


Figure 7. Palestrina.



NOTES

[1]Porter, Andrew. Musical Events. *The New Yorker*, Vol. LXIII, No.12, 11 May 1987, p. 101. This is not to disparage the intellectual ferment that the *Journal of Music Theory* represents. That publication has been a standard-bearer in the struggle to have music theory recognized as a serious academic discipline. Without it, Paul Harder, Walter Piston and other such "cookbook writers" might still be considered insightful theorists.

[2]Two highly interesting papers appeared in *College Music Symposium*, 17/1, addressing similar issues. One is by Allen Forte, "Music Theory in Re-Transition: Centripetal Signs" (pp. 156-162) and the other, by Carl Schachter, "Diversity and the Decline of Literacy in Music Theory" (pp.150-153).

[3]Harder, Paul. *Harmonic Materials in Tonal Music*, Fifth Edition, Parts One and Two. Allyn and Bacon, 1985. Although this work went through numerous editions, their differences are remarkably slight. An example of its musical (not to mention intellectual) softheadedness can be gained from this (sadly typical) extract: "Borrowed chords have the same harmonic function as their unaltered counterparts. A

subdominant triad, for example, tends to progress to the dominant, tonic, or supertonic chords regardless of whether it is major or minor in quality. Since the use of borrowed chords does not affect harmonic function, what is the reason for using them? (Answer:) For greater tonal variety or color." (Part Two. Frame 380, p. 152.)

Ironically, the notion of programmed texts, a vogue of the 1960s, arose from the rudimentary "teaching machines," ancestors of the "educational" computer, then being widely discussed. It is a credit to the maturity of much computer-based instruction that it has progressed far beyond the mere automation of programmed texts to the point of discarding their approach.

[4]The interested reader is referred to an excellent article by Alexander R. Brinkman, "Representing Music Scores For Computer Analysis" in the *Journal of Music Theory*, 30/2 (Fall 1986), pp. 225-273.

[5]Jeppesen, Knud. *Counterpoint*. Translated by Glenn Haydon. Prentice-Hall, 1939.

[6]Jeppesen, p. 138.

[7]Piston, Walter and Mark DeVoto. *Harmony*. Fifth edition. W. W. Norton, 1986.

[8]The downloadable "Sonata" font is available from Adobe Systems, Mountain View, CA 94039, (415)-962-2100. The reader is directed to Vol. 2, No. 7 (16 February 1988) of *MacWEEK*, a weekly trade magazine devoted to the Macintosh, pp. 34-39 for a discussion of it and for a general introduction to the Musical Instrument Digital Interface on the Macintosh.

[9]*Melodious Dictator*, designed by David B. Williams. Available from Temporal Acuity Products, Inc., Bellevue, WA 98005. (206)-746-2790.